WHAT IS CLAIMED IS:

1. A method of driving the simulation testing of a design of an integrated circuit (IC) which is to be incorporated into an intended system comprising the steps of:

providing an asynchronous sequence of states configured for simulating operating conditions relevant to driving sequencing of signal-exchange events with said IC;

identifying first upper and first lower parameters of timing constraints imposed by said intended system with respect to enabling individual said events;

forming a first synchronous sequence of states in which said states are synchronized on a basis of remaining within said first upper and first lower parameters of timing constraints;

identifying second upper and second lower parameters of timing constraints imposed by said IC with respect to enabling individual said events;

forming a second synchronous sequence of states in which said states are synchronized on a basis of remaining within said second upper and said second lower parameters of timing constraints; and

using said second synchronous sequence as a basis for said simulation testing of said design.

2. A method of generating a synchronous sequence of test vectors from information originating within an asynchronous environment comprising:

providing a simulation synchronous sequence of states, wherein each of said states is referenced to a clock period, said simulation synchronous sequence being partially based on event timing parameters of a particular system of interest;

introducing short timing delays to said states within specific said clock periods of said simulation synchronous sequence to generate an asynchronous short-delay sequence of states, durations of specific said short timing delays being responsive to event timing parameters of a particular integrated circuit (IC) design;

comparing said states of said asynchronous short-delay sequence, including correlating a plurality of said clock periods having said states of said asynchronous short-delay sequence to identify a first overlapping time interval, said first overlapping time interval being consistent with a time coincidence among said states of said asynchronous short-delay sequence;

introducing long timing delays to said states within specific said clock periods of said simulation synchronous sequence to generate an asynchronous long-delay sequence of states, durations of specific said long timing delays being responsive to event timing parameters of said particular IC;

comparing said states of said asynchronous long-delay sequence, including correlating a plurality of said clock periods having said states of said asynchronous long-delay sequence to identify a second overlapping time interval, said second overlapping time interval being consistent with a time coincidence among said states of said asynchronous long-delay sequence;

generating a synchronous long-delay sequence by successively repeating a second delay-adjusted clock period having a state which is delayed by said second overlapping time interval; and

comparing said synchronous short-delay sequence with timing of said states of said synchronous long-delay sequence to generate said synchronous sequence of test vectors, including time aligning said synchronous short-delay and long-delay sequences to detect a plurality of overlapping sampling time intervals for locating said synchronous sequence of test vectors.

3. The method of claim 2 wherein said step of introducing said short timing delays includes adding best case tester-load timing delays to said clock periods of said simulation synchronous sequence, said best case tester-load timing delays being indicative timing constraints of an IC tester.

4. The method of claim 2 wherein said step of introducing said short timing delays includes adding best case chip-load timing delays indicative of timing constraints of said IC design.

5. The method of claim 3 wherein said step of introducing said long timing delays includes adding worst case tester-load timing delays that are indicative of said timing constraints of said IC tester.

6. The method of claim 4 wherein said step of introducing said long timing delays includes adding worst case chip-load timing delays indicative of said timing constraints of said IC.

7. The method of claim 2 wherein said step of providing said simulation synchronous sequence includes:

providing a simulated asynchronous sequence of states;

extracting a state of said asynchronous sequence at each said clock period to generate a simulated synchronous sequence of states;

introducing an abbreviated timing delay to each said clock period of said simulated synchronous sequence to generate a simulated synchronous abbreviated-delay sequence and introducing an extended timing delay to each said clock period of said simulated synchronous sequence to generate a simulated synchronous extended-delay sequence; and

comparing said simulated synchronous abbreviated-delay sequence to said simulated synchronous extended-delay sequence, including time aligning said simulated synchronous abbreviated-delay and extended-delay sequences to detect a plurality of overlapping second time intervals for defining positions of states in said clock periods of said simulation synchronous sequence.

8. The method of claim 7 wherein said step of introducing said abbreviated timing delay and said extended timing delay includes executing said simulated synchronous sequence under respective best case timing delay and worst case timing delay scenarios in a system simulation environment, said system simulation environment having timing characteristics indicative of said particular system of interest.

9. The method of claim 7 further including adapting said simulated synchronous extended-delay sequence as said simulation synchronous sequence when there is not an acceptable number of said overlapping second time intervals.

10. The method of claim 7 wherein said step of providing said simulated asynchronous sequence includes selecting said clock period to have a duration that corresponds to a tester clock period of an IC tester.

11. The method of claim 2 further including selectively fixing a sampling instance in one of said overlapping sampling time intervals to correspond to a rising edge of a tester clock period of an IC tester.

12. A test vector generator for generating a synchronous sequence of test vectors comprising:

a simulation module that is enabled to generate a simulation synchronous sequence of states under a system simulation environment, said simulation synchronous sequence including a plurality of timing regions for identifying operations of an integrated circuit (IC) design;

a delay module that is enabled to introduce short delays and long delays to said simulation synchronous sequence to respectively generate asynchronous short-delay sequence and asynchronous long-delay sequence, each of said short delays and said long delays being timing delays associated with at least one of an integrated circuit (IC) and an IC tester;

an overlaying module that is configured to provide a first state overlapping time interval and a second state overlapping time interval by respectively comparing a plurality of base periods of said asynchronous short-delay sequence and comparing a plurality of base periods of said asynchronous long-delay sequence;

a duplication module that is configured to incorporate said first state time interval into a first sequence of said base periods and to incorporate said second state overlapping time interval into a second sequence of said base periods to respectively generate a synchronous short-delay sequence and a synchronous long-delay sequence; and

a sequence overlaying module that is configured to time align said synchronous short-delay sequence and said synchronous long-delay sequence to detect a plurality of overlapping sampling intervals for locating said synchronous sequence of test vectors.

13. The test vector generator of claim 12 wherein said short delays are related to a best case chip-load timing delay of said IC and a best case tester-load timing delay of said IC tester.

14. The test vector generator of claim 13 wherein said long delays are related to a worst case chip-load timing delay of said IC and a worst case tester-load timing delay of said IC tester.

15. The test vectors generator of claim 12 further comprising a verification module that is configured to execute said synchronous sequence of test vectors under said short delays and said long delays for verifying timing correctness.

16. The test vectors generator of claim 12 wherein said system simulation environment is independent of any delay associated with said IC and said IC tester.

17. The test vectors generator of claim 12 wherein said base period is a time interval that is equivalent to a tester period of said IC tester.

18. A method for converting asynchronous states into synchronous states to generate a synchronous sequence of test vectors for verifying functionality of a simulated integrated circuit (IC) design comprising:

providing a simulation synchronous sequence of states;

generating an asynchronous short-delay sequence of first periods and an asynchronous long-delay sequence of second periods, including inserting short delays and long delays into said simulation synchronous sequence, said short delays and said long delays characterizing timing delays of at least one of said simulated IC and a tester;

detecting a short-delay overlapping time interval and a long-delay overlapping time interval, including correlating a plurality of said first periods to identify said short-delay overlapping time interval and correlating a plurality of said second periods to identify said long-delay overlapping time interval;

generating a synchronous short-delay sequence of states by forming a succession of substantially identical base periods that include a state and said short-delay overlapping time interval;

generating a synchronous long-delay sequence of states by forming a succession of substantially identical base periods that include a state and said long-delay overlapping time interval; and

generating said synchronous sequence of test vectors, including time-aligning said synchronous short-delay sequence and said synchronous long-delay sequence and identifying overlapping timing envelopes of states within corresponding said base periods of said synchronous short-delay and long-delay sequences.

19. The method of claim 18 wherein said step of inserting said short delays and said long delays includes respectively introducing best-case timing delays of said IC and worst-case timing delays of said IC.

20. The method of claim 19 wherein said step of inserting said short delays and said long delays includes respectively introducing best-case timing delays of said tester and worst-case timing delays of said tester.